

MOTION FILED  
DEC 7 1995

(24)  
No. 94-2003

**In the  
Supreme Court of the United States  
October Term, 1995**

**Lotus Development Corporation,**  
*Petitioner,*

**v.**

**Borland International, Inc.,**  
*Respondent.*

**ON WRIT OF CERTIORARI TO THE  
UNITED STATES COURT OF APPEALS  
FOR THE FIRST CIRCUIT**

**MOTION FOR LEAVE TO FILE BRIEF AMICUS CURIAE  
AND BRIEF AMICUS CURIAE OF SOFTWARE PROTECTION  
COMMITTEE OF MINNESOTA INTELLECTUAL PROPERTY  
LAW ASSOCIATION IN SUPPORT OF RESPONDENT**

**Steven W. Lundberg\***  
**Bradley A. Forrest**  
**Danjal J. Kluth**  
**Schwegman, Lundberg &  
Woesener, P.A.**  
**80 S. Eighth Street, #3500**  
**Minneapolis, MN 55402**  
**(612) 373-6902**

**Rudolph P. Hofmann Jr.**  
**Westman, Champlin, Kelly,  
P.A.**  
**121 S. Eighth Street, #720**  
**Minneapolis, MN 55402**  
**(612) 334-3222**  
**Nelson R. Capes**  
**Palmatier, Sjoquist  
& Helget, P.A.**  
**7900 Xerxes Ave. S., #2000**  
**Minneapolis, MN 55431**  
**(612) 831-5454**

*Counsel for Amicus Curiae*

*\*Counsel of Record*

IN THE  
**SUPREME COURT OF THE UNITED STATES**

OCTOBER TERM, 1995

---

No. 94-2003

---

LOTUS DEVELOPMENT CORPORATION,  
*Petitioner,*

v.

BORLAND INTERNATIONAL, INC.,  
*Respondent.*

---

**On Writ of Certiorari to the  
United States Court of Appeals  
for the First Circuit**

---

**MOTION FOR LEAVE TO FILE BRIEF AMICUS  
CURIAE**

---

The Software Protection Committee of the Minnesota Intellectual Property Law Association, Inc., 514 Nicollet Mall, Suite 300, Minneapolis, MN 55401, ("MIPLA"), a regional association of primarily attorneys, whose interest and practices lie in the areas of copyright, patent, trademark, trade secret and other intellectual property law, respectfully moves for leave to file the attached brief *amicus curiae* in this copyright infringement case. The consent of the attorney for respondent has been obtained. Consent of the attorney for

the petitioner had not been obtained at the time this motion was printed.

The Software Protection Committee of MIPLA has over forty members and offers the views of a leading regional group of intellectual property attorneys concerned with copyrights and patents for software, representing a broad range of views. The Software Protection Committee of MIPLA therefore respectfully requests that the Court grant its motion for leave to file the accompanying brief, and consider the Software Protection Committee's arguments during its deliberations.

The attached brief was not reviewed by MIPLA as a whole, and represents only the views of the Software Protection Committee of MIPLA.

Respectfully submitted,

STEVEN W. LUNDBERG\*

CHAIR OF AD HOC SUBCOMMITTEE OF  
SOFTWARE PROTECTION COMMITTEE  
MINNESOTA INTELLECTUAL  
PROPERTY LAW ASSOCIATION, INC.  
3500 IDS Center  
80 South Eighth Street  
Minneapolis, MN 55402  
(612) 373-6902

NELSON R. CAPES, BRADLEY A. FORREST,  
RUDY P. HOFMANN, JR. & DANIEL J. KLUTH  
*Counsel for Amicus Curiae*

December 4, 1995

\* Counsel of Record

TABLE OF CONTENTS

	Page
TABLE OF AUTHORITIES .....	iii
INTEREST OF AMICUS CURIAE .....	1
SUMMARY OF ARGUMENT .....	2
ARGUMENT .....	3
I.    SECTION 102(B) EXCLUDES ELECTRONIC OPERATION FROM COPYRIGHT PROTECTION .....	3
II.   COMPUTER PROGRAMS AND ELECTRONIC OPERATION ARE SEPARATE AND DISTINCT .....	7
III.  BORLAND IS FREE TO PROVIDE A COMPUTER HAVING AN IDENTICAL ELECTRONIC OPERATION AS LOTUS 1-2-3 .....	10
IV.  THE LOTUS 1-2-3 MENU COMMAND HIERARCHY CHARACTERIZES ELECTRONIC OPERATION, NOT A PROTECTIBLE ASPECT OF THE LITERARY FORM OF A COMPUTER PROGRAM .....	11
V.    SOFTWARE COPYRIGHTS VS. SOFTWARE PATENTS: RELATIVE RISK AND RETURN .....	16
A.    LENGTH OF PROTECTION .....	16
B.    RIGHTS PROTECTED .....	16



C.	COST AND DIFFICULTY OF OBTAINING PROTECTION .....	17
D.	FRINGE BENEFITS .....	18
E.	COPYRIGHT FOR <i>DE FACTO</i> STANDARDS .....	20
VI.	CONCLUSION .....	22

## TABLE OF AUTHORITIES

Cases	Page
<i>Apple Computer, Inc. v. Microsoft Corp.</i> , No. 93-16883 (9th Cir. Sept. 19, 1994) .....	11
<i>Ashton-Tate Corp. v. Ross</i> , 728 F.Supp 597 (D. N. Cal. 1989), <i>aff'd</i> 916 F.2d 516 (1990) .....	11
<i>Baker v. Selden</i> , 101 U.S. 99 (1879) .....	5
<i>Digital Communications Assoc. v. Softklone Distrib. Corp.</i> , 659 F.Supp 449 (D. N. Ga. 1987) .....	11
<i>Dred Scott v. Sandford</i> , 60 U.S. (19 How.) 393, 564 (1857) (Curtis, J., dissenting) .....	7
<i>Feist Publications Inc. v. Rural Telephone Serv. Co.</i> , 499 U.S. 340 (1991) .....	3
<i>Lemelson v. United States</i> , 752 F.2d 1538, 224 U.S.P.Q. (BNA) 526 (Fed. Cir. 1985) .....	17
<i>Lotus Dev. Corp. v. Borland Int'l Inc.</i> , 49 F.3d 807 (1st Cir. 1995) .....	10, 13
<i>Mazer v. Stein</i> , 347 U.S. 201 (1954) .....	5
<i>NEC Corp. v. Intel Corp.</i> , 10 U.S.P.Q.2d (BNA) 1177 (1989) .....	9, 11
<i>Pennwalt Corp. v. Durand-Wayland, Inc.</i> , 833 F.2d 931, 3 U.S.P.Q.2d (BNA) 1737 (Fed. Cir. 1987) ( <i>en banc</i> ) .	17
<i>Philbrook v. Glodgett</i> , 421 U.S. 707 (1975) .....	4
<b>Constitutional Provisions, Statutes and Public Laws</b>	
17 U.S.C. § 101 .....	5
17 U.S.C. § 102 .....	18
17 U.S.C. § 102(b) .....	2, 3, 4, 5, 6, 7, 9
17 U.S.C. § 302 .....	16
35 U.S.C. § 101 .....	18
35 U.S.C. § 102 .....	18
35 U.S.C. § 103 .....	18
35 U.S.C. § 112 .....	17, 20
35 U.S.C. § 154 .....	16

37 C.F.R. § 1.20(c) .....	19
37 C.F.R. § 1.20(e),(f),(g) .....	17
37 C.F.R. § 1.510 .....	19
37 C.F.R. § 202.20(c)(vii)(2) .....	20
37 C.F.R. § 202.3(c) .....	18
Pub. L. No. 93-573, § 201(b)-(c), 88 Stat. 1973-74 (1974) .....	4

### Legislative Materials

H.R. Rep. No. 1476, 94th Cong., 2d Sess. 57 (1976) reprinted in 1976 U.S. Code Cong. and Admin. News 5659 .....	4
--	---

### Other Authorities

Gene K. Landry, <i>Macros and Broad Software Copyright Protection: The Latest Round in Lotus v. Borland</i> , Computer Law., Vol. 11, No. 2, Feb. 1994 .....	21, 22
Steven W. Lundberg, George H. Gates III, and John P. Sumner, <i>Baker v. Selden, Computer Programs</i> , 17 U.S.C. § 102(b) and <i>Whelan Revisited</i> 13 Hamline L. Rev. 221 (1990) .....	6, 9
Gregory J. Maier, <i>Software Protection - Integrating Patent, Copyright and Trade Secret Law</i> , 69 J. Patent and Trademark Off. Soc'y 151 (1987) .....	6
<i>Final Report of the Nat'l Comm. On Technological Uses of Copyrighted Works</i> (1978) .....	4
M. Nimmer, 3 Nimmer on Copyright § 13.01[B] .....	20
Reichman, <i>Computer Programs As Applied Scientific Know- How: Implications of Copyright Protection For Commercialized University Research</i> , 42 Vand. L. Rev. 639 (1989) .....	5
E. Sergeant, <i>Justice Touched with Fire</i> , in Mr. Justice Holmes 206-07 (F. Frankfurter ed. 1931) .....	7

Frederick R. Warren-Boulton, et. al., <i>POINT: Copyright Protection of Software Can Make Economic Sense</i> , Computer Law., Vol. 12, No. 2, Feb. 1995 .....	21, 22
--	--------

IN THE  
**SUPREME COURT OF THE UNITED STATES**

OCTOBER TERM, 1995

---

No. 94-2003

---

LOTUS DEVELOPMENT CORPORATION,  
*Petitioner,*

v.

BORLAND INTERNATIONAL, INC.,  
*Respondent.*

---

On Writ of Certiorari to the  
United States Court of Appeals  
for the First Circuit

---

**BRIEF AMICUS CURIAE OF  
SOFTWARE PROTECTION COMMITTEE  
OF THE MINNESOTA INTELLECTUAL  
PROPERTY LAW ASSOCIATION, INC.  
IN SUPPORT OF RESPONDENT**

---

**INTEREST OF AMICUS CURIAE**

The Software Protection Committee of the Minnesota Intellectual Property Law Association, Inc., 514 Nicollet Mall, Suite 300, Minneapolis, MN 55401, ("MIPLA") files this *amicus curiae* brief to aid the Court in its review of the ruling of the United States Court of Appeals for the First Circuit in the above-titled case.



The Software Protection Committee of MIPLA offers the views of a leading regional group of intellectual property attorneys concerned with copyrights and patents for software, representing a broad range of views. The Software Protection Committee of MIPLA has over forty members, many of them with extensive practical experience in applying copyright and patent law to protect computer software. Attorney members of Software Protection Committee represent both copyright owners and their competitors. Accordingly, the Software Protection Committee's opinion is the product of a non-partisan analysis. This brief has not been reviewed by MIPLA as a whole, and represents only the views of the Software Protection Committee.

MIPLA is a regional association of primarily attorneys, whose interest and practices lie in the areas of copyright, patent, trademark and other intellectual property law. Membership in MIPLA is drawn from private law firms, corporations and universities.

### SUMMARY OF ARGUMENT

The Copyright Act of 1976 expressly excludes from copyright protection any idea, procedure, process, system, method of operation, concept, principle or discovery.<sup>1</sup> A computer program and the electronic operation of a computer executing the program are separate and distinct. While copyright clearly extends to a computer program as a literary work, Congress intended that electronic operation be excluded from protection under Section 102(b). Characteristics of electronic operation can be separated from the characteristics of the literary form of a computer program. The former can be excluded from protection under Section 102(b) without directly impinging on the protection of the latter.

The menu command hierarchy Lotus seeks to protect specifies how an electronic spreadsheet device will respond to a set

---

<sup>1</sup> 17 U.S.C. § 102(b).

of electronic symbols<sup>2</sup> when applied to the device in a defined syntax. This hierarchy constitutes choices of electronic design which were antecedent to the coding of the Lotus 1-2-3 computer program, and are separate and distinct therefrom. Moreover, the hierarchy, disembodied from a specific literary form, characterizes the operation of an electronic device or computer and not the literary form of a computer program or user-interface, and as such is uncopyrightable. To the extent a computer program or user interface must embody a "copy" of the electronic symbols and/or syntax logic of the menu command hierarchy in order to achieve the electronic operation defined thereby, then such commands and/or syntax are necessary to the electronic operation of, or part of a method of operating, a computer, and are therefore uncopyrightable. The First Circuit was correct, therefore, in finding that the menu command hierarchy of Lotus 1-2-3 was uncopyrightable under 17 U.S.C. § 102(b).

### ARGUMENT

#### I. Section 102(b) Excludes Electronic Operation from Copyright Protection

The Copyright Act of 1976 expressly excludes from copyright protection any idea, procedure, process, system, method of operation, concept, principle or discovery. 17 U.S.C. § 102(b). Section 102(b) identifies those elements of a work, otherwise deserving of copyright protection, for which copyright is not available. *Feist Publications Inc. v. Rural Telephone Serv. Co.*, 499 U.S. 340, 356 (1991). Additionally, the legislative history of Section 102(b) makes it clear that Congress intended to exclude the

---

<sup>2</sup> These digital symbols are translatable to English language letters which have a verbal relationship to the electronic operation invoked by the symbols when supplied to a computer. The letters themselves, apart from their associated digital symbols, have no meaning to the computer.

electronic operation<sup>3</sup> of a computer, as opposed to the mode of expression, from software copyright protection.<sup>4</sup> In particular, the House Report on the 1976 Copyright Act sets forth:

Section 102(b) is intended, among other things, to make clear that the expression adopted by the programmer is the copyrightable element in a computer program, and that the actual processes or methods embodied within the program are not within the scope of copyright law.

H.R. Rep. No. 1476, 94th. Cong., 2d Sess. 57 (1976) *reprinted in* 1976 U.S.C.C.A.N. 5659, 5660.

The idea/expression dichotomy with respect to software was also given treatment in the 1978 Report of the National Commission on New Technology Uses of Copyrighted Works (CONTU) preceding the 1980 amendment to the Copyright Act.<sup>5</sup> Following extensive study, CONTU sought to clarify the distinction between copyrightable computer programs and uncopyrightable electronic operation when it stated that copyright law protects the program as long as it remains fixed in a tangible medium of expression, but not the electro-mechanical functioning of a machine. *See, Final Report of the Nat'l Comm. On Technological Uses of Copyrighted Works*, at 48-49 (1978). Consequently, the definition of "computer program" as recommended by CONTU and

<sup>3</sup> As used herein, the term "electronic operation" refers to the electro-mechanical functioning of a computer and to electronic systems, methods, or processes carried out on a computer.

<sup>4</sup> Legislative intent governs decisions on the construction of statutes. *E.g., Philbrook v. Glodgett*, 421 U.S. 707, 713 (1975).

<sup>5</sup> Congress established CONTU in part to study the relationship between copyright protection and computer technology, and to make recommendations to amend the copyright law where appropriate. Pub. L. No. 93-573, § 201(b)-(c), 88 Stat. 1973-74 (1974).

set forth in the 1980 amendment to the Copyright Act is simply "a set of statements or instructions to be used directly or indirectly in a computer to bring about a certain result". 17 U.S.C. § 101.

The Supreme Court's first comprehensive review of the idea/expression dichotomy occurred in *Baker v. Selden*, 101 U.S. 99 (1879), later restated in *Mazer v. Stein*, 347 U.S. 201, 217 (1954), which established the principles subsequently codified in § 102(b). The Court in *Baker* held that the plaintiff's copyright of a textbook describing a new method of accounting did not grant plaintiff a monopoly on the accounting system and that "writings" necessary to use the accounting system were *per se* uncopyrightable. *Id.* at 103. The Court's analysis turned on the proposition that copyright in a writing should not be used to indirectly obtain protection to utilitarian works more properly the subject of patent protection.<sup>6</sup> To merely state that *Baker*, and subsequently § 102(b), asserts a generalized idea/expression dichotomy is to gut the meaning therein. The idea/expression dichotomy existed since the earliest origin of copyright law, and well before *Baker*.<sup>7</sup>

The Court in *Baker* did not apply an idea/expression analysis of the traditional type. Rather, the Court in *Baker* permitted the defendant's reproduction of materials otherwise copyrightable under a traditional idea/expression analysis, in order to facilitate the use by the defendant of the unprotectible utilitarian features of the accounting system. *Id.* at 104. The *Baker* Court also held that the explanatory essay was protected by copyright, but not infringed. *Id.* Viewed as such, the analysis under *Baker*, and its corresponding provision § 102(b), requires defining the unprotected utilitarian feature, and then determining whether

<sup>6</sup> *Id.* at 103.

<sup>7</sup> *See, Reichman, Computer Programs As Applied Scientific Know-How: Implications of Copyright Protection For Commercialized University Research*, 42 Vand. L. Rev. 639, 693 n.288 (1989).



copyright protection for the corresponding writing would affect the right of others to practice the feature.

In the context of computer programs, the statutory mandate requires drawing a distinction between the electronic operations performed by a computer and the set of statements or instructions of a computer program.<sup>8,9</sup> In other words, Congress limited the definition of copyright protection available to a computer program to its literary form, i.e. its "statements and instructions", to exclude the electronic operation of a computer from copyright protection. Thus, to the extent that electronic operation can be excluded from protection while maintaining protection for the literary form, the Court has received a statutory mandate to do so.

In the past, the plain meaning of section 102(b) has been ignored by many courts, contrary to the intent of Congress, presumably to fill in the gaps left when courts denied patents for software inventions. See, Gregory J. Maier, *Software Protection - Integrating Patent, Copyright and Trade Secret Law*, 69 J. Patent and Trademark Off. Soc'y 151, 161 (1987). Stretching copyright protection in order to protect a programmer's valuable efforts may be understandable or even equitable, but if it is not within the confines of the statutory mandate, it is not acceptable.<sup>10</sup> The

---

<sup>8</sup> See Steven W. Lundberg, George H. Gates III, and John P. Sumner, *Baker v. Selden, Computer Programs*, 17 U.S.C. § 102(b) and *Whelan Revisited* 13 Hamline L. Rev. 221, 228 (1990).

<sup>9</sup> To help keep the distinction between electronic operation and computer programs clear within this brief, the term "computer program" is only used to refer to the literary form of a computer program, and not to the electronic operation effectuated by the computer program when it is executed in a computer.

<sup>10</sup> When a strict interpretation of a statute, according to the fixed rules which govern the interpretation of laws, is abandoned, and the theoretical opinions of individuals are allowed to control its meaning, we no longer have the statute; we are under the government of individuals, who for the time being have power to declare what the statute is, according to their own personal views of what it ought

decisions ignoring or misapplying the idea/expression dichotomy intended by Congress proceed from sympathy rather than law.<sup>11</sup> As such, Section 102(b) instructs the Court to cover only the literary form of a computer program, and to exclude from protection the electronic operation of computers.

## II. Computer Programs and Electronic Operation are Separate and Distinct

As demonstrated below, the literary form of a computer program is separable from the electronic operation resulting from the execution of the program in a computer. Characteristics of literary form, as that term is used herein, include the literal sequence of program code,<sup>12</sup> the structure, sequence and organization of the code, and literary or audiovisual elements of a user-interface resulting from the execution of the code in a computer.

Characteristics of the literary form of program code do not include characteristics of electronic operation resulting from the execution of the code in a computer. Characteristics of electronic operation include, *inter alia*, electronic functions<sup>13</sup> or processes for transforming data from one state to another, the electronic flow of

---

to mean. See, *Dred Scott v. Sandford*, 60 U.S. (19 How.) 393, 564 (1857) (Curtis, J., dissenting).

<sup>11</sup> It is reported that Justice Holmes and Judge Learned Hand once had lunch together and afterward, as Justice Holmes began to leave in his carriage, Judge Hand ran after him shouting, "Do justice sir, do justice!" Justice Holmes stopped the carriage and rebuked Justice Hand: "That is not my job. It is my job to apply the law." E. Sergeant, *Justice Touched with Fire*, in Mr. Justice Holmes 206-07 (F. Frankfurter ed. 1931).

<sup>12</sup> As used herein, the term "code" refers to the "statements or instructions" of a computer program.

<sup>13</sup> Such as moving a column of numbers in a spreadsheet, or storing data in a file.

data through a computer, and the control or invoking of electronic functions, processes, or data flow by commands or data supplied to the computer by a user or external device. The lexicon of electronic characteristics is independent of whether the electronic function is "hard-wired" (*i.e.*, no software) or implemented with a programmable device.

Electronic operation or functionality is separable from the literary form of program code when a multiplicity of computer programs with dissimilar literary forms<sup>14</sup> can be used to achieve the desired electronic operation or functionality.<sup>15</sup> In such cases, the literary form of any one of these dissimilar computer programs can be granted generous copyright protection without precluding others from attaining the same electronic operation using dissimilar code.

The electronic operation resulting from a given computer program, when executed in a computer, can be defined by electronic characteristics. If there exists, for the purpose of obtaining electronic operation having the defined electronic characteristics, a multiplicity of computer programs of dissimilar literary form which are plausible alternatives to one another, and if the defined characteristics do not intrinsically suggest one alternative form over the other, then it can be said that the defined characteristics do not provide "access"<sup>16</sup> to the given computer

---

<sup>14</sup> Dissimilar literary forms are those that are not substantially similar, that is they do not use the same literal sequences of code, nor the same structure, sequence or organization of code.

<sup>15</sup> In cases of merger of literary form with electronic operation, none of the literary forms are protectible because there are few or only one form capable of effectuating the desired electronic function.

<sup>16</sup> This term is used in its sense as an element of proof of copyright infringement.

program, and are separable therefrom.<sup>17</sup> This principle has even been applied to separate the characteristics of electronic operation from the literary form of machine code,<sup>18</sup> and can be used for user-interfaces.<sup>19</sup>

A defined set of electronic characteristics which do not provide access to the literary form of an associated computer program can be said to be non-informing.<sup>20</sup> Any set of non-informing electronic characteristics can therefore be treated as *per se* uncopyrightable under 17 U.S.C. § 102(b) without impinging on copyright protection for the literary form of a computer program. These non-informing characteristics belong to the universe of processes, systems and/or methods of operation, as those terms were intended by Congress to apply to exclude protection to the "electro-mechanical functioning of a machine."

It follows from the above that infringement of a computer program cannot be established on the basis of copying non-informing electronic characteristics alone. In all cases, infringement must be based on the copying of the literary form of

---

<sup>17</sup> An example of a "non-informing" definition of electronic characteristics is the user manual of Lotus 1-2-3. This manual describes the operation of the Lotus 1-2-3 electronic spreadsheet but, with the exception of the description of the literary form of the user-interface, does not describe in any way the literary form of the computer program Lotus uses to obtain this operation.

<sup>18</sup> *NEC Corp. v. Intel Corp.*, 10 U.S.P.Q.2d (BNA) 1177 (1989). Machine code has a very close correspondence to electronic operation.

<sup>19</sup> It is noted that by definition a user-interface provides access and control over the electronic operation obtained by the associated computer program. There is therefore a reasonable likelihood that many elements of a user-interface will be necessary to use of or part of the method of operation of such uncopyrightable electronic operation, and hence uncopyrightable under 17 U.S.C. § 102(b).

<sup>20</sup> For an example of this principle, see Steven W. Lundberg, George H. Gates III, and John P. Sumner, *Baker v. Selden, Computer Programs*, 17 U.S.C. § 102(b) and *Whelan Revisited* 13 Hamline L. Rev. 221, 22-230 (1990).



a computer program, and not on similarity of electronic operation. This distinction, fundamental to the proper application of copyright law to computer software, has been lost on a number if not most courts.<sup>21</sup> In the instant case, the First Circuit properly recognizes the distinction between a computer program and electronic operation.<sup>22</sup>

Because the above-recited principles can be used to cleanly reverse-engineer uncopyrightable electronic operation from the protectible literary form of a computer program, electronic operation can be freely excluded from copyright protection without direct effect on the protection of the literary form of computer programs.<sup>23</sup>

### III. Borland is Free to Provide a Computer Having an Identical Electronic Operation as Lotus 1-2-3.

Under current copyright law in all the Circuits, provided one does not copy the literal code, the look and feel, the sequence, structure and organization, nor the filtered protectible elements of the Lotus 1-2-3 computer program, one is free to duplicate the electronic operation realized by running the program on a suitable

<sup>21</sup> This is most evident in the uninformed manner in which many prior decisions on copyright protection for software use the term computer program in two different senses: in one sense the term is used to refer to an inert literary work, and in another sense to the electronic operation resulting from the program when it is executed in a computer. These uninformed mixed uses of this term have muddled the distinction between literary form and electronic operation.

<sup>22</sup> The First Circuit refers to the electronic form simply as "Lotus 1-2-3," and does not confuse a computer program with electronic operation. *Lotus Dev. Corp. v. Borland Int'l Inc.*, 49 F.3d 807, 809 (1st Cir. 1995).

<sup>23</sup> Of course, the literary form of a computer program will be subject to other exclusions from copyright, such as uncopyrightable ideas, *scenes a faire* and merger.

computer.<sup>24</sup> This electronic operation can include the exact same electronic functions as Lotus 1-2-3 without worry of copyright infringement. Many cases have so held under the idea versus expression analysis.<sup>25</sup>

### IV. The Lotus 1-2-3 Menu Command Hierarchy Characterizes Electronic Operation, not a Protectible Aspect of the Literary Form of a Computer Program

The menu command hierarchy of the Lotus 1-2-3 spreadsheet system at issue in this case is a command set that has a defined syntax or logical system relating the commands to one another for the purpose of controlling an electronic spreadsheet system. Each of the commands, when used as part of a macro language, take the form of unique electronic binary symbols.<sup>26</sup>

<sup>24</sup> Computers may provide electronic function with no programming at all. In fact, most of the first computers were special purpose machines designed to provide electronic operations such as the adding of two numbers. Software, as we know it today had not yet been invented. Electrical circuitry and logic gates provided the ability to accept numbers, modify them and provide an output. Software was invented to make the computers easy to modify to perform new tasks or electronic operations. It is very possible to provide a hardware-only machine to implement an electronic spreadsheet that appears to operate identical to the Lotus 1-2-3 electronic spreadsheet system running on a programmable computer. Copyright cannot be used to prevent making such a machine and should not be used to prevent use of it.

<sup>25</sup> E.g., see *NEC Corp. v. Intel Corp.*, 10 U.S.P.Q.2d (BNA) 1177 (1989), *supra*, *Apple Computer, Inc. v. Microsoft Corp.*, No. 93-16883 (9th Cir. Sept. 19, 1994), *Ashton-Tate Corp. v. Ross*, 728 F.Supp. 597 (D.N. Ca. 1989), *aff'd* 916 F.2d 516 (1990), *Digital Communications Assoc., Inc. v. Softklone Distrib. Corp.*, 659 F.Supp. 449 (D. N. Ga. 1987).

<sup>26</sup> Actual literary forms of these commands exist only in the form of a visualization of the menu command hierarchy in the user interface of the Lotus 1-2-3 electronic spreadsheet system, in a displayed or printed macro file, or in the documentation for the system. Each of the binary symbols are chosen from a set of electronically encoded ASCII alpha-numeric characters (one symbol for each character of the alphabet and the numbers 1-10, as well as for many other



These binary symbols, used in certain combinations as prescribed by the rules of the syntax, are provided to a computer or other electronic device to invoke electronic operations such as copying a column or row of a displayed spreadsheet.<sup>27</sup>

In order to provide electronic operation in accordance with the commands and syntax of the menu command hierarchy of the Lotus 1-2-3 spreadsheet system, an electronic device must recognize these commands and the syntax of their use. If the electronic device is hard-wired (i.e. contains no software), the circuits are designed to recognize the binary symbols of the commands and respond to them in accordance with the syntax, which is also encoded in the form of electronic logic.<sup>28</sup> Copyright cannot be used to prevent making such a machine and should not be used to prevent use of it. If the electronic device is a programmable computer, the electronic binary symbols and the syntax logic are embodied in a computer program used to obtain the desired electronic operation.

Whether one implements the menu command hierarchy of the Lotus 1-2-3 electronic spreadsheet system in a hard-wired device or a programmable one, the disembodied<sup>29</sup> menu command hierarchy is nothing but a specification of uncopyrightable

---

common punctuation marks or abbreviations), so that the symbol also represents a letter of the alphabet that has a verbal relationship to the command.

<sup>27</sup> In essence, as stated by the First Circuit, the commands are like electronic buttons that are part of the method of operating the Lotus 1-2-3 electronic spreadsheet.

<sup>28</sup> This is not only theoretically possible, it is technically quite easy to achieve.

<sup>29</sup> "Disembodied" is used here in the sense of being divorced from any particular embodiment in the literary form of a computer program.

electronic operation.<sup>30</sup> Furthermore, the disembodied symbols and syntax logic of the Lotus 1-2-3 electronic spreadsheet system constitute choices of electronic design which were antecedent to the coding of the Lotus 1-2-3 computer program,<sup>31</sup> and were intended from the start to constitute characteristics of electronic operation.

The central point is that Lotus is not entitled to copyright a set of electronic symbols and syntax logic that they have made necessary to the operation of an electronic device.<sup>32</sup> The Court should not be confused by the fact that these symbols also represent letters of the English alphabet (according to the ASCII standard) that describe or suggest the electronic operation invoked by the symbols. In point of fact, these letters themselves have no meaning to a computer; what Lotus really seeks is to prevent Borland from programming a computer to map the electronic symbols and syntax logic of the Lotus 1-2-3 menu command hierarchy to the corresponding electronic operation of Borland's spreadsheet.

To the extent the Lotus 1-2-3 computer program embodies a "copy" of the electronic symbols and/or syntax logic of the hierarchy, the symbols and their syntax logic constitute a preexisting work merely incorporated in the program to provide a

---

<sup>30</sup> It is particularly plain to see that the syntax constitutes electronic characteristics, as it defines how a computer will respond to different combinations of inputs, has no value or use outside of defining this response, and because it is not readily seen how this syntax is embodied in literary form in a computer program.

<sup>31</sup> See *Lotus Dev. Corp. v. Borland Int'l Inc.*, 49 F.3d 807, 816 (1st Cir. 1995) (Speaking of the underlying computer code being different from the menu command hierarchy).

<sup>32</sup> If Lotus can do this, then the electronic symbols used for internal messaging and control by other kinds of electronic devices also should be copyrightable, provided they are chosen from a set of ASCII symbols having the pretense of a literary work. This would, in effect, extend copyright protection to all digital electronics.

new work of authorship constituting a compilation or collection of this preexisting work and the code of the computer program.<sup>33</sup> As such, the commands and their syntax cannot correctly be characterized as non-literal aspects of the program,<sup>34</sup> even if both are covered by the same copyright registration.

In fact, if the disembodied menu commands and their syntax *per se*<sup>35</sup> are not considered part of the Lotus 1-2-3 computer program, it is clear that the disembodied menu command hierarchy which Lotus seeks to protect is both separable from and non-informing with respect to the literary form of the code of the Lotus 1-2-3 computer program.<sup>36</sup> As such, as established in Section III above, Lotus cannot maintain that the code of its Lotus 1-2-3 computer program has been infringed on the basis of copying the disembodied menu command hierarchy.

To establish a case of infringement of the Lotus 1-2-3 computer program, Lotus must at a minimum show that Borland copied some aspect of the manner in which the menu command

---

<sup>33</sup> Even if the menu command hierarchy was not "fixed" prior to the Lotus 1-2-3 computer program, it would still be separable therefrom as electronic characteristics. This case presents the somewhat unusual case where the codes used for electronic operation also correspond by translation to a "classic" literary form.

<sup>34</sup> That these commands and syntax are embodied in some literary form in the Lotus 1-2-3 computer program does not transform them from uncopyrightable electronic characteristics to copyrightable non-literary elements of a computer program. By that reasoning otherwise uncopyrightable telephone listings could be made copyrightable simply by incorporating them in a computer program.

<sup>35</sup> *I.e.*, not of any particular literary form in the program.

<sup>36</sup> More specifically, there is no way to know from this disembodied hierarchy what literary form a computer program will take for the purpose of embodying the hierarchy. There are a multitude of different ways the commands and syntax of the Lotus 1-2-3 spreadsheet system can be embodied in a computer program.

hierarchy is embodied in the Lotus 1-2-3 computer program or the user-interface of the Lotus 1-2-3 spreadsheet system. Here, Lotus has done neither.

As pointed out above, the disembodied menu command hierarchy of the Lotus 1-2-3 spreadsheet system, standing alone, clearly constitutes uncopyrightable characteristics of the electronic operation. Furthermore, to the extent it is necessary to embody some literary form of the menu command hierarchy into a computer program to obtain the uncopyrightable electronic operation defined in the hierarchy, then the embodiment of that literary form is necessary to the use of the electronic operation,<sup>37</sup> and therefore unobjectionable under copyright law.<sup>38,39</sup> Accordingly, the First Circuit was correct to exclude the menu

---

<sup>37</sup> Or, as put by the First Circuit, the menu command hierarchy is a part of the method of operating the uncopyrightable spreadsheet system.

<sup>38</sup> It is noted again that Lotus has not alleged that Borland has copied any particular literary form of the menu command hierarchy, but only that it has been necessarily copied in some form, and that they are entitled to protect all forms, however embodied, under copyright. It is noted that the visualization or literary form of the user interface, and the program code itself of the Lotus 1-2-3 computer program may be copyrightable as literary works provided they may be expressed in many ways, but they are not the subject of this appeal. The question before this court is whether the disembodied menu command hierarchy, which when implemented on a computer and allows a user to control the Lotus 1-2-3 electronic spreadsheet system, is copyrightable in and of itself.

<sup>39</sup> As Borland has implemented commands in the hierarchy, a user does not see anything of the user interface that users of a computer running Lotus 1-2-3 see. Rather, the commands are translated or interpreted in real time into commands that are usable by a computer running the Borland Quattro program. A table of user input commands, comprising the first letter of the user commands is used to make that translation. When a command is received, the computer searches down the table to identify the correct Quattro code to execute. No organization of the independent commands was copied from Lotus' organization of the code corresponding to such commands other than that necessarily incident to implement sets of hierarchically related commands, such as move a row or column.



command hierarchy of Lotus 1-2-3 from copyright protection.

## V. Software Copyrights vs. Software Patents: Relative Risk and Return

The treatment of software copyrights by some courts, such as the *Lotus* district court, result in a larger universe of protected rights at significantly lower cost than would result from a corresponding United States patent. This does not appear to be sound policy, nor in keeping with Congress' intent, as noted above, to limit the scope of copyright protection for software. Set forth below is a comparison of patent and copyright protection for computer programs.

### A. Length of Protection

Patents on applications filed after June 8, 1995, have a term of protection that begins on the date of grant and ends twenty years from the filing date of the application for the patent.<sup>40</sup> Copyright in a work created after January 1, 1978 generally endures for a term consisting of the life of the author plus fifty years.<sup>41</sup> Clearly, a software copyright has a term much longer than a corresponding software patent.

### B. Rights Protected

A United States patent grants to the patentee, his heirs or assigns, the right to exclude others from making, using, or selling the invention throughout the United States.<sup>42</sup> The "invention"

---

<sup>40</sup> 35 U.S.C. § 154.

<sup>41</sup> 17 U.S.C. § 302.

<sup>42</sup> 35 U.S.C. § 154.

refers to the invention literally set forth in the claims<sup>43</sup> plus substantial equivalents.

In order to infringe a patent, a device must include each and every limitation literally set forth in at least one patent claim, or the substantial equivalent thereof.<sup>44,45</sup> Patent claims thus provide a very good guide to the extent of protection provided by the patent. If even one claimed element is missing from an accused device, the accused device will not literally infringe the patent.<sup>46</sup>

In contrast, a copyright registration application has no such property definition. There is thus little certainty for a competitor who wishes to develop a competitive product: until the copyright owner brings suit for infringement, the developer cannot know which aspect of the computer program, i.e., source code, object code, structure, sequence and organization, user interface, or parameter lists the copyright owner may assert as infringed.

### C. Cost and Difficulty of Obtaining Protection.

A United States patent typically costs the patentee between \$5,000.00 to \$25,000.00 to obtain. Thereafter, at three, seven, and eleven years after issue, maintenance fees of \$990.00, \$1990.00, and \$2990.00 are due.<sup>47</sup> An application for copyright application

---

<sup>43</sup> 35 U.S.C. § 112.

<sup>44</sup> *Lemelson v. United States*, 752 F.2d 1538, 224 U.S.P.Q. (BNA) 526 (Fed. Cir. 1985).

<sup>45</sup> *Pennwalt Corp. v. Durand-Wayland, Inc.*, 833 F.2d 931, 3 U.S.P.Q.2d (BNA) 1737 (Fed. Cir. 1987) (en banc).

<sup>46</sup> Thus, the claims of a software patent perform the very important function of notifying competitors of what they may and may not make, sell and use without infringement.

<sup>47</sup> 37 C.F.R. § 1.20(e),(f),(g).



for a single work must be accompanied by a fee of \$20.00.<sup>48</sup> No other administrative or maintenance fees are required during the term of the copyright.

The difficulty of obtaining a U.S. patent is also enormously greater than of obtaining a copyright registration. To obtain a U.S. patent, a patentee must show that the invention is novel,<sup>49</sup> non-obvious,<sup>50</sup> and useful.<sup>51</sup> The novelty and non-obvious barriers are difficult to surmount and generally require extensive prosecution by the patent attorney.<sup>52</sup>

In contrast, the copyright statute only requires "original" works of authorship.<sup>53</sup> There is no requirement for novelty, non-obviousness, or utility. In particular, a copyright author need not demonstrate that he was the first to create the particular expression embodied in the work.

#### **D. Fringe Benefits**

Not only is a software copyright cheaper to obtain in relation to the extent of rights protected than a software patent, but a software copyright owner also enjoys a number of additional

---

<sup>48</sup> 37 C.F.R. § 202.3(c).

<sup>49</sup> 35 U.S.C. § 102.

<sup>50</sup> 35 U.S.C. § 103.

<sup>51</sup> 35 U.S.C. § 101 ("Whoever invents or discovers any new and useful . . .").

<sup>52</sup> In addition, for software inventions, there is the "mathematical algorithm" bar to patentability. This is a court-developed (and PTO-developed) doctrine that has generally prevented the grant of a patent to software inventions which are mere mathematical algorithms on the grounds that they do not meet statutory grounds for patentability under 35 U.S.C. § 101. This doctrine is a formidable barrier to patentability of software inventions.

<sup>53</sup> 17 U.S.C. § 102.

benefits, many of them relating to infringement, that a patent owner does not enjoy.

First, a software copyright is much less susceptible to a challenge of invalidity than a software patent. It is a virtual certainty that if a software patentee sues an alleged infringer, the alleged infringer will counterclaim that the patent is invalid, either on the grounds that it is anticipated or obvious over the prior art, or that "inequitable conduct" was used in procuring the patent. Although an alleged copyright infringer may also challenge the copyright on the grounds that it does not contain copyrightable subject matter, the standard for copyrightability is originality, which is much less difficult to uphold than the novelty standard of patentability. The copyright owner thus has substantially less risk in attempting to enforce his exclusive rights than does the patentee.

Furthermore, a competitor need not manufacture a competitive device and take a chance on being sued for patent infringement. For a fee, the competitor may request that the Patent and Trademark Office re-examine the issued patent on the basis of prior art patents or printed publications which the competitor believes invalidate the patent.<sup>54</sup> There is no similar mechanism available to a competitor concerned about a software copyright: he must essentially produce the competing product and take his chances in court. Thus, the owner of the software copyright has much more surety of maintaining his monopoly.

The proofs required for copyright infringement are a mixed blessing for the copyright owner. He is disadvantaged relative to the patentee in that the copyright owner must prove that the alleged infringer copied the work, while the patentee need not do so (innocent infringement is not a defense in a patent infringement suit). However, copying may be proved by circumstantial evidence, usually on the basis that the alleged infringer had access

---

<sup>54</sup> 37 C.F.R. § 1.510; 37 C.F.R. § 1.20(c).

to the allegedly copied work.<sup>55</sup> In software copyright infringement, especially for mass-distributed software, this would not appear to be a difficult problem.

On the other hand, the patentee must, in order to prove literal infringement, show that each and every element and limitation of at least one patent claim is present in the accused device. A competitor can sometimes readily "design around" a patent by making slight modifications.

Finally, the software copyright owner, but not the patentee, may maintain important aspects of his work as trade secrets while still enjoying a statutory monopoly. The Copyright Act allows the copyright owner to block out significant portions of the work deposited with the Copyright Office.<sup>56</sup> In contrast, the patentee must disclose the "best mode" contemplated by the inventor of carrying out his invention.<sup>57</sup> If the patentee attempts to keep the best mode a trade secret, his patent may later be held invalid.

In summary, the software copyright owner arguably enjoys a larger universe of protected rights, with additional fringe benefits, at enormously lower cost of procurement than the software patentee.

#### E. Copyright for *De Facto* Standards

A recent article presents a very forceful economic argument that copyright protection of *de facto* standards should not be

<sup>55</sup> M. Nimmer, 3 Nimmer on Copyright § 13.01[B].

<sup>56</sup> 37 C.F.R. § 202.20(c)(vii)(2).

<sup>57</sup> 35 U.S.C. § 112.

granted.<sup>58</sup> The article argues that granting copyright protection to *de facto* standards limits the "network externalities" which occur "when the value of a product or service to a buyer increases with the cumulative number of other buyers".<sup>59</sup>

Decisions like *Lotus v. Borland* make it less likely that the network benefits of standard user interfaces will be realized. Lotus 1-2-3 users, who have invested thousands of dollars in training on the standard 1-2-3 user interface, will have only one source for a spreadsheet program that uses this standard interface, i.e., Lotus 1-2-3. As is described in the Landry article,<sup>60</sup> the "slippery slope" of this decision may lead to a "broadening of copyright protection for program command protocols without logical limit."<sup>61</sup> This may have enormous consequences in the areas of program-to-program communication, modems, telecommunications, and computer languages.<sup>62</sup>

Furthermore, the problem of software copyright protection of non-literal program elements extends well beyond *de facto* standards, as has been discussed above. The point is not that Lotus, or any developer of a computer program, should be denied all protection for that program, but that the scope of protection awarded should be commensurate with the cost of obtaining that protection and competitors should have clear notice, *ex ante*, of those aspects which are being protected.

<sup>58</sup> Frederick R. Warren-Boulton, et. al., *POINT: Copyright Protection of Software Can Make Economic Sense*, Computer Law., Vol. 12, No. 2, Feb. 1995 (hereinafter "Warren-Boulton").

<sup>59</sup> *Id.* The example given is that of the telephone network.

<sup>60</sup> Gene K. Landry, *Macros and Broad Software Copyright Protection: The Latest Round in Lotus v. Borland*, Computer Law., Vol. 11, No. 2, Feb. 1994.

<sup>61</sup> Landry, *supra*, at n.39.

<sup>62</sup> *Id.*



Warren-Boulton et. al. argue that:

the uncritical, unexamined, and extensive nature of protection under copyright is efficient only when the expected welfare loss of its being granted mistakenly is *de minimis*. Like summary judgment and the *per se* rule, copyright is the cost-effective approach only when the probability of error (i.e., overprotection) is very small, and/or when the potential social loss associated with the error is very small.<sup>63</sup>

A significant probability of overprotection may exist anytime copyright protection is given to non-literal program elements in the realm of electronic operation. This is the case because extending copyright protection to these elements results in greater protection than afforded by a patent at enormously lower cost.

## VI. Conclusion

In the computer program copyright paradigm Lotus seeks the Court to adopt, electronic operation is indistinct from and in fact merged with the literary form of a computer program. Thus, according to Lotus, it is not possible to separate protectible writing from electronic operation, and precluding protection to the latter *per se* implicitly violates the intent of Congress to protect the former. This is not the case and Lotus' position should be rejected.

In its holding, the First Circuit found, in essence, that the menu command hierarchy was a part of the electronic operation of Lotus 1-2-3, and was therefore uncopyrightable. The First Circuit's decision respects the paradigm sought by Congress in which electronic operation is excluded from copyright protection,

---

<sup>63</sup> *Id.*

while at the same time permitting liberal copyright protection to the literary form of computer programs. Accordingly, the decision should be upheld.

Respectfully submitted,

STEVEN W. LUNDBERG (Counsel of Record)  
CHAIR OF AD HOC SUBCOMMITTEE OF  
SOFTWARE PROTECTION COMMITTEE  
MINNESOTA INTELLECTUAL  
PROPERTY LAW ASSOCIATION, INC.  
3500 IDS Center  
80 South Eighth Street  
Minneapolis, MN 55402  
(612) 373-6902

NELSON R. CAPES, BRADLEY A. FORREST,  
RUDY P. HOFMANN, JR. & DANIEL J. KLUTH  
*Counsel for Amicus Curiae*  
December 4, 1995